10. ročník Žilina 2. – 4. 7. 2018
Otvorený softvér vo vzdelávaní, výskume a v IT riešeniach
osscconf.soit.sk
OSS Conf

OSSConf 2026: 1–7

Několik postřehů k Asymptote

PAVEL STŘÍŽ (CZ)

Abstrakt. Článek je stručný přehled možností, jak získat 3D výstupy z R za pomoci Asymptote a TEXu.

Klíčová slova. TEX, Asymptote, R.

SEVERAL NOTES ON ASYMPTOTE

Abstract. The article is a brief research of possibilites of 3D outputs from R via Asymptote and TEX.

Keywords. TEX, Asymptote, R.

1. Vzpomínky na Asymptote

Na své první pokusy s Asymptote si pamatuji dobře, neb jsem řešil tři školské grafy ve 3D paralelně v Asymptote a přes Jmol v prostředí Sage. Od té doby již neužíváme balíček movie15, ale jeho nástupce balíček media9.

```
$ texdoc movie15 media9
```

Inspirací ke studiu mi byla hlavně tato galerie, http://www.piprime.fr/asymptote/. Autorem ukázek je Philippe Ivaldi.

2. Instalace Asymptote

Poněvadž pracuji na starších strojích, vždy něco nejede, byť používám poslední verzi TEXLive. V tomto případě se mi nepodařilo generovat pdf skrz starší grafickou kartu, až po zadání

```
export MESA_GL_VERSION_OVERRIDE=1.2
```

do souboru ~/.bashrc a načtení přes source ~/.bashrc se mi to podařilo, ale to jen pro obrázky vykreslené ve 2D. U hledání problému mi pomohly parametry Asymptote -v, -vv (nebo -v -v) a hlavně -vvv (nebo -v -v -v). Ve výpisech lze jít až do páté úrovně (str. 168 dokumentace). Linuxový man hovoří o -v a jisté možnosti jít hlouběji, ale nehovoří konkrétně.

```
$ texdoc asy # nebo
$ man asy
```

Přesedl jsem na novější stroj. Na Xubuntu 20.04 bylo potřeba doinstalovat knihovnu freeglut3, speciálně se jednalo o knihovnu libglut.so.3.

```
$ sudo apt install freeglut3
```

Poté se mi podařilo zobrazit všechny grafy od Rudolfa Blaška.

3. WebGL

Zajímavá je i situace, když chceme získat webovou verzi grafů. Možná si říkáte, proč je to tak důležité. Pokud pracujete v linuxovém prostředí, tak firma Adobe zrušila podporu pro prohlížeč pdf Adobe Reader. Poslední známá verze je 9.5.5 pro 32bitové stroje. Uživatelé pak musí sáhnout po virtuálním stroji typu Oracle VM VirtualBox, po Wine 32bitové verzi, míchat 32bitové a 64bitové aplikace či zkusit podobné řešení (FoxIt Reader, ale také jede jen něco). Stručně řečeno, prakticky na libovolném stroji s Microsoft Windows či MacOS 3D grafy interaktivně v pdf uvidíte, ale zápasíte s tím pod Linuxem.

O WebGL (vhodné i pro chytré telefony) a nastavení je zmínka na str. 137 v dokumentaci Asymptote. Poslední verze programu je 2.65.

```
$ texdoc asymptote
```

Vzal jsem si jeden ze vzniklých asy souborů Rudolfa Blaška, zkopíroval na `webgl-pokus.asy` a vyčistil do této minimální podoby (vymazání úvodních definic, globálního definování neužitých barev a promazání prázdných řádků):

```
import graph3;
import palette;
size(175,175,IgnoreAspect);
currentprojection=perspective(2,.55,0.25);
real f(pair z) {return sin(z.x)*sin(z.y);}
draw(surface(f,(-pi,-pi),(pi,pi),nx=24,Spline),lightblue,render(merge=true));
surface f=surface(f,(-pi,-pi),(pi,pi),24,Spline);
draw(f,mean(palette(f.map(zpart),Rainbow(24))),brown);
draw(Label("$x$",1),(-1.2pi,0,0)--(1.2pi,0,0),brown,Arrow3);
draw(Label("$y$",1),(0,-1.2pi,0)--(0,1.2pi,0),brown,Arrow3);
draw(Label("$z$",1),(0,0,-1.2)--(0,0,1.2),brown,Arrow3);
label("$z=\sin{x}\cdot\sin{y}$",(.5,.5,-1.35),SE,deepblue);
```

O výstupech hovoří manuál v kapitole 8.29 `three`, speciálně na str. 137. Prvně jsem si vyzkoušel náhled v okně:

```
$ asy -V webgl-pokus.asy
```

Nasledované pokusem o vygenerování webové stránky.

```
$ asy -f html webgl-pokus.asy
```

Vzniká soubor `webgl-pokus.html`, který se mi podařilo otevřít i na starém notebooku. Alternativou je zadat výstupní formát přímo v asy souboru (středník na konci řádků je nutný, je součástí syntaxe jazyka):

```
import settings;
settings.outformat="html";
// outformat="html";
```

Bez prázdných řádků soubor vypadá přibližně takto:

```
<!DOCTYPE html>
<!-- Use the following line to embed this file within another web page:
<iframe src="webgl-pokus.html" width="176" height="176" frameborder="0"></iframe>
-->
<html lang="">
<head>
  <title>webgl-pokus</title>
  <meta http-equiv="content-type" content="text/html; charset=utf-8"/>
  <meta name="viewport" content="user-scalable=no"/>
  <script
  src="https://vectorgraphics.github.io/asymptote/base/webgl/asygl-1.00.js">
  </script>
  <script>
    canvasWidth=176;
    canvasHeight=176;
    absolute=false;
    // ...
    Centers=[
    [-10.3373,-38.23629,-223.1933],
    [81.93387,-1.42227,-289.569],
    [0.5386997,67.84467,-262.8985],
    [9.891595,-74.96387,-327.4771],
    ];
  </script>
</head>
<body style="overflow: hidden;" onload="webGLStart();">
  <canvas id="Asymptote" width="176" height="176" style="border: none;">
  </canvas>
</body>
</html>
```

Nepodařilo se mi dle návodu na str. 137 vložit vzniklou webovou stránku do další přes <object> ani přes <iframe>, jak radí poznámka ve vzniklém souboru. V konzoli se píše SecurityError: Permission denied to access property "document" on cross-origin object.

Jisté řešení nabízí odpovědi na webové stránce https://stackoverflow.com/questions/36333978/error-permission-denied-to-access-property-document, ale to je bezpečnostní riziko. Vyčištěním vzniklého souboru na úroveň prostého JavaScriptu a přes <canvas> se mi sice pořadilo dostat jeden graf na webovou stránku, ale jen jeden. Ideální je zůstat u hypertextových odkazů na vznikající soubory dle vzoru galerií na https://asymptote.sourceforge.io/gallery/.

Pro badatele jistě stojí za pozornost možnost konverze PostScriptu do formátu asy s možností náhledu a následné editace:

```
$ pstoedit -f asy <soubor.eps> <soubor.asy>
$ asy -V <soubor.asy>
```

V případě problémů s písmy se u `pstoedit` používá parametr `-dt`.

4. Ovládání TEXové proměnné mimo TEX

Rudolf Blaško byl spokojený s prvními výstupy, ale u jeho knihy bylo potřeba předchozí kroky zautomatizovat, jednalo se o definice v prostředí `asydef`, které bylo potřeba změnit. Zdá se, že ani parametry z příkazového řádku programu `asy` tyto definice nemění. Chceme totiž knihu nechat beze změny, ale zároveň si vygenerovat webové stránky s 3D objekty.

Vyřešil jsem to přepínačem na TEXové úrovni, který řídí dávkový soubor. Ukažme si minimální ukázku. Tento blok jsem si přidal za autorovy definice.

```
\newcount\prepinac
\input ovladani.tex
\ifnum\prepinac=1
\begin{asydef}
settings.outformat="html";
settings.inlineimage=false;
settings.embed=false;
\end{asydef}
\fi
```

V dávkovém souboru pak stačí mít něco takového.

```
soubor="ovladani.tex"
# Získání pdf knihy...
echo "\\prepinac=0" >$soubor
lualatex kniha.tex # plus další příkazy
# Generování html s 3D objekty...
echo "\\prepinac=1" >$soubor
lualatex kniha.tex # plus další příkazy
```

Tento přístup se používá, když chceme vygenerovat několik verzí knihy, např. knihu ve více šablonách, verzi s barevnými a černobílými obrázky, verzi na tisk a na obrazovku, verzi s a bez 3D objektů, pracovní a konečnou verzi časopisu ap.

5. Generování asy souborů přes R: jednoduchá ukázka

O interaktivní 3D grafice jsem poprvé četl ve Zpravodaji CSTUGu od Romana Plcha, viz `https://www.cstug.cz/bulletin/pdf/2013-1.pdf`. Tam je zmínka o generování asy souborů, včetně jazyka R, konkrétně se jednalo o knihovnu `misc3d` zmíněnou na str. 40. Knihovna umí exportovat objekt `Triangles3D`, který se musí získat. Na knihovně staví novější knihovna `plot3D`.

Zvláštní kategorii tvoří knihovna `rgl`. Přišlo mně zajímavé vyzkoušet export do asy, tím lze získat 3D model v pdf, ale zároveň interaktivní webovou stránku (WebGL), proto nepoužiji příkaz `writeWebGL`, ale `writeASY`. Zkusme si minimální ukázku a v tomto článku se zaměřím už jen na webovou stránku.

Připravíme si soubor `test-rgl.R`:

```
# install.packages("rgl")
library(rgl)
with(iris, plot3d(Sepal.Length, Sepal.Width, Petal.Length, type="s",
     col=as.numeric(Species)))
writeASY()
```

Spustíme:

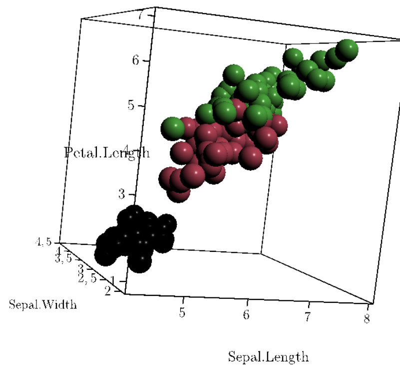
```
$ Rscript test-rgl.R
```

S největší pravděpodobností dostaneme chybovou hlášku od programu Asymptote. Psal jsem autorovi, chyba je opravena, ale ještě nová verze není dostupná ve všech linuxových a TEXových distribucích. Nepomůže nám ani pokus přes `writeASY(ver244=TRUE)`.

Vzniká nám soubor `scene.asy`, který upravíme. Do druhého řádku zasáhneme: `settings.outformat = "html";` a vymažeme 5 řádků, od `currentlight = light(` po nejbližší `);`.

Spouštíme následující řádky a dostáváme takový výsledek.

```
$ asy -v scene.asy\
$ firefox scene.html
```



Badatele odkazuji na prozkoumání stránky `https://cran.r-project.org/web/packages/rgl/vignettes/rgl.html`.

6. Generování asy souborů přes R: pokročilejší ukázka

V IB č. 1/2018 jsem na str. 13–15 zmínil jednoduché ukázky z manažerského rozhodování, z oblasti rozhodování při riziku. Chtěl bych si zkusit obdobné grafy, ale interaktivní. Představím vám knihovnu gMOIP, resp. mohu doporučit webové stránky autora (`https://github.com/relund/gMOIP`) a jeho blog (`https://www.research.relund.dk/`).

Naše problémy jsou, že budeme chtít víc 3D modelů a určitě nebudeme chtít dělat ruční zásahy. Ukážeme si zásahy do textových souborů přes Lua, je to jazyk založený na C++, který dominuje v TEXovém světě a hraje nemalou roli ve světě

počítačových her. Speciální pozornost by si zasloužila knihovna LPeg, která si získala místo ve světě regulárních výrazů.

Připravíme si soubor `test-gmoic.R`:

```
# install.packages("gMOIP")
library(gMOIP)
# definice
A <- matrix( c(3, 2, 5,  2, 1, 1,  1, 1, 3,  5, 2, 4), nc = 3, byrow = TRUE)
b <- c(55, 26, 30, 57)
obj <- c(20, 10, 15)
view <- matrix( c(-0.45, -0.446, 0.77, 0, 0.886, -0.32, 0.33, 0, 0.098, 0.835, 0.54, 0,
     0, 0, 0, 1), nc = 4)
# první obrázek
loadView(v = view, zoom = 0.75)
plotPolytope(A, b, faces = c("c","c","c"), type = c("c","i","i"), plotOptimum = TRUE, obj
     = obj)
rgl::writeASY(outtype = "pdflatex", prc = TRUE, title = "polytope-1")
# druhý obrázek
loadView(v = view, zoom = 0.75)
plotPolytope(A, b, type = c("c","c","i"), plotOptimum = TRUE, obj = obj, plotFaces =
     FALSE)
rgl::writeASY(outtype = "pdflatex", prc = TRUE, title = "polytope-2")
```

V dalším kroku si připravíme TEXový dokument `asy-pres-r.tex`:

```
\documentclass[a4paper]{article}
\usepackage[luatex]{graphicx}
\usepackage{asymptote}
\begin{document}
\input{polytope-1.tex}\par
\input{polytope-2.tex}
\end{document}
```

Následuje soubor v Lua, který nám upravuje získané soubory, nazval jsem jej `smaz-light.lua`:

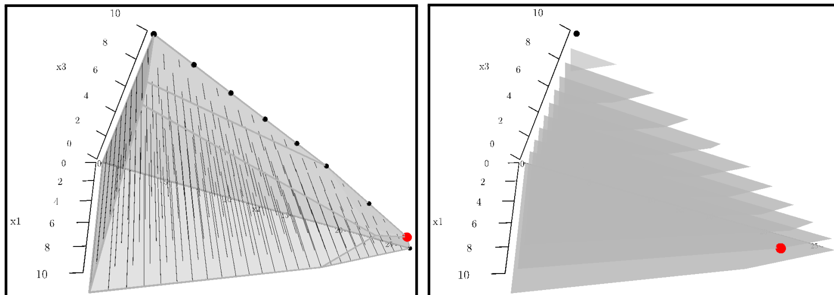
```
-- Načtení souboru
nazev=arg[1]
print("Zpracovávám soubor "..nazev)
soubor=io.open(nazev)
obsah=soubor:read("*all")
soubor:close()
-- Zásah do souboru
obsah=unicode.utf8.gsub(obsah,"currentlight =.-%);", "", 1)
obsah=unicode.utf8.gsub(obsah, "settings%.inlineimage=true;",
     "settings.inlineimage=false;", 1)
-- Přepsání původního souboru
kam=io.open(nazev,"w")
kam:write(obsah)
kam:close()
```

Nyní nezbývá než si připravit dávkový soubor `davka.sh`, který spustíme přes `bash davka.sh`:

```
# Základ generování asy souborů
zaklad="asy-pres-r"
Rscript test-gmoic.R
lualatex --shell-escape $zaklad.tex 1>/dev/null 2>&1
# Úpravy v asy souborech
for soubor in `find -iname $zaklad-\*.asy`; do
texlua smaz-light.lua $soubor
asy -v $soubor
asy -v -f html $soubor
done
# Náhled na výsledky
firefox $zaklad-*.html
```

Co se přesně děje? Z R se vygenerují dva TeXové soubory, které použije TeX a při prvním běhu vygeneruje asy soubory. Do nich zasáhneme na textové úrovni a přes Asymptote vygenerujeme 3D objekty a webové stránky, které si otevřeme ve webovém prohlížeči. Výhoda tohoto přístupu je, že u dalších běhů LuaLaTeXu vysázíme v `asy-pres-r.pdf` 3D modely. Vedle toho máme dostupnou interaktivní webovou verzi.

Zde je náhled na očekávaný výsledek ve dvou záložkách prohlížeče.



Kontaktní adresa

Ing. Pavel Stříž, Ph.D., U Škol 940, Bučovice, okres Vyškov, 685 01, Česká republika,
E-mailová adresa: `pavel@striz.cz`